# MixupE: Understanding and Improving Mixup from Directional Derivative Perspective

Yingtian Zou*, Vikas Verma*
joint work with
*Sarthak Mittal, Wai Hoh Tang, Hieu Pham, Juho Kannala, Yoshua Bengio, Arno Solin, Kenji Kawaguchi*

*National University of Singapore* *Aalto University*
*Universite de Montreal, Mila* *Google Brain*

Conference on Uncertainty in Artificial Intelligence, 1st Aug 2023

# Outline

# Backgrounds

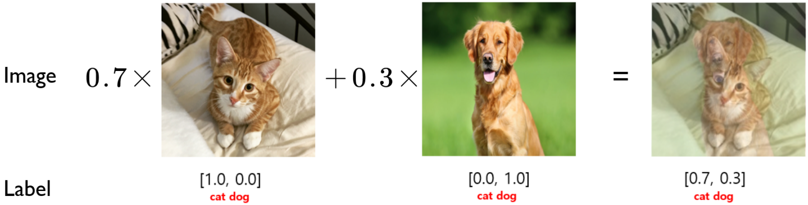



**Figure:** Mixup for Image Classification

Modeling the uncertainty of in-between samples.

# Mixup formulation

With coefficient $\lambda \sim \mathrm{Beta}(\alpha, \alpha)$, for $\lambda \in [0, 1], \alpha \in (0, \infty)$.
Mixup generates a virtual in-between sample,

$$\tilde{x} = \lambda x_i + (1 - \lambda) x_j,$$
$$\tilde{y} = \lambda y_i + (1 - \lambda) y_j,$$

where $(x_i, y_i)$ and $(x_j, y_j)$ are two feature-target vectors drawn at random from the training data.

The mixup hyper-parameter $\alpha$ controls the strength of interpolation between feature-target pairs, recovering the Empirical Risk Minimization (ERM) principle as $\alpha \to 0$.

# Smoother feature space

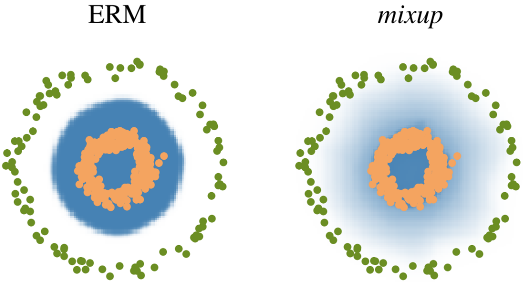
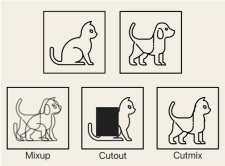


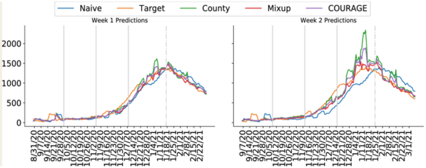

**Figure:** Illustrative sample referred from [Zhang et al., 2018]. The green and orange dots represent different classes. Blue shading indicates the probability $p(y = 1|x)$. Mixup yields a smoother decision boundary in feature space than ERM.

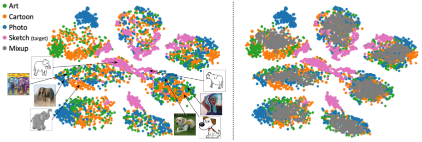
# Applications

Mixup now has been widely applied to various areas, including

- Image classification/ generation
- Out-of-Distribution/Domain Generalization
- Node and graph classification
- Time Series Prediction .....

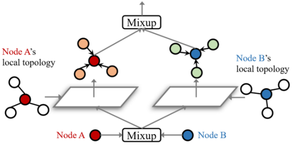



(a) Original Latent Distribution

(b) Mixup by Random Interpolation




Our two-branch Mixup graph convolution

# Outline

# Implicit Regularization

Implicit Regularization, also referred to as *Implicit Bias*, characterizes the underlying term to be optimized when training an algorithm.

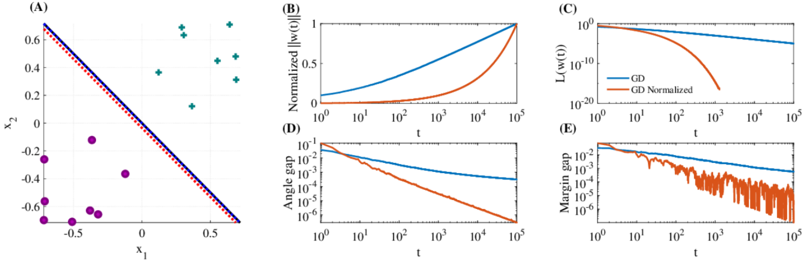



**Figure:** [Soudry et al., 2018] show the implicit bias (margin maximization) of Gradient Descent (GD) on binary classification with logistic regression.

# Implicit Regularization of Mixup

- Activated feature:

$$h(f_\theta(\mathbf{x})) = \begin{cases} \log\left(\sum_j \exp(f_\theta(\mathbf{x})_{(j)})\right) & \text{Softmax} \\ \log\left(1 + \exp\left(f_\theta(\mathbf{x})\right)\right) & \text{Sigmoid} \end{cases}$$

- Loss function: $\ell(\theta, (\mathbf{x}, \mathbf{y})) = h(f_\theta(\mathbf{x})) - \mathbf{y}^\top f_\theta(\mathbf{x})$
- Mixup data: $\tilde{\mathbf{x}}_{i,j}(\lambda) = \lambda \mathbf{x}_i + (1-\lambda)\mathbf{x}_j$, and $\tilde{\mathbf{y}}_{i,j}(\lambda)$
- Mixup loss:

$$L_n^{\text{mix}}(\theta, S) := \frac{1}{n^2} \sum_{i,j=1}^{n} \mathop{\mathbb{E}}_{\lambda \sim \text{Beta}(\alpha,\beta)} l(\theta, \tilde{\mathbf{x}}_{i,j}(\lambda), \tilde{\mathbf{y}}_{i,j}(\lambda))$$

# Implicit Regularization of Mixup

**Theorem 1**

Let $a_\lambda = 1 - \lambda$, $\ell(\theta, (\mathbf{x}, \mathbf{y})) \triangleq h(f_\theta(\mathbf{x})) - \mathbf{y}^\top f_\theta(\mathbf{x})$ be the loss function and $\forall \theta \in \Theta$ functions $f_\theta(\cdot)$ in a $C^K$ manifold. Then the implicit regularization of Mixup is:

$$L_n^{\text{mix}}(\theta, S) = L_n^{\text{std}}(\theta, S) + R$$

$$R = \frac{1}{n} \sum_{i=1}^{n} \mathbb{E}_{\substack{\lambda \sim \mathcal{D}_\lambda \\ \mathbf{x}' \sim \mathcal{D}_X}} \left( \sum_{k=1}^{K} \frac{a_\lambda^k}{k!} \mathbf{J}_h^k(f_\theta) \Delta_i^{\otimes k} - a_\lambda \mathbf{y}_i^\top \Delta_i + a_\lambda^K \hat{\psi}_{i,\mathbf{x}'}(a_\lambda) \right)$$

where $\mathbf{J}_h(f_\theta)(\mathbf{x}_i) = g(f_\theta(\mathbf{x}_i))^\top$ and

$$\Delta_i = \sum_{k=1}^{K} \frac{a_\lambda^{k-1}}{k!} \mathbf{J}_{f_\theta}^k(\mathbf{x}_i)(\mathbf{x}' - \mathbf{x}_i)^{\otimes k} + a_\lambda^{K-1} \psi_{i,\mathbf{x}'}(a_\lambda).$$

# Implication of Theorem 1

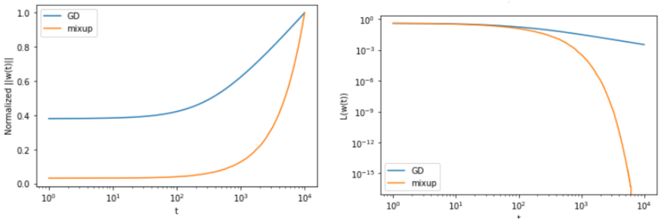
1. Minimizing mixup loss is equivalent to adding an implicit regularization $R$ to ERM loss.
2. $R$ mainly depends on the directional derivatives, since $\hat{\psi}_{i,\mathbf{x}'}$ and $\psi_{i,\mathbf{x}'}$ are the remainder terms in Taylor expansion of order $\mathcal{O}(K)$ and with probability 1,

$$\lim_{a_\lambda \to 0} \hat{\psi}_{i,\mathbf{x}'}(a_\lambda) = 0, \quad \lim_{a_\lambda \to 0} \psi_{i,\mathbf{x}'}(a_\lambda) = 0.$$

3. The function $f_\theta(\cdot)$ should be at least twice continuously differentiable.

# Toy example: Linear Logistic Binary Classification

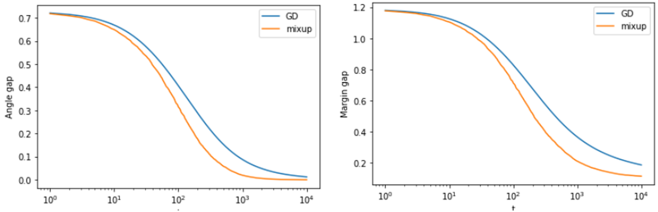
We follow [Soudry et al., 2018] to conduct an experiment on separable data.




(a) Normalized $w(t)$ (b) Training loss

**Figure:** On a linear model $f_w$, training $f_w$ with ERM or Mixup yields the same implicit bias (loss decreases, norm of $w(t)$ explodes). In other words, the implicit bias of Mixup vanishes on the linear model.

# Same implicit bias on Linear Model




(a) Angle gap (b) Margin gap

**Figure:** Implicit bias of binary classification with logistic regression on the linear model. From the results we can see, both Mixup and GD are maximizing the margin and have similar convergence rates.

# Conclusion

- Minimizing mixup loss is equivalent to adding an implicit regularization to ERM loss.
- The implicit regularization has a complicated form.

# Outline

# Limitation

[✗] Minimizing the implicit regularization of Mixup in Theorem 1 explicitly is impractical.

[✓] Retaining Mixup with an extra regularization is a computationally efficient alternative way.

[✗] Using high-order approximations suffers a heavy computational burden in deep learning.

[✓] Regularize model with only first-order (dominate) approximation.

# Proposed MixupE

The first-order directional derivative is captured by

$$D^1_{\theta,S} := \frac{1}{n}\mathbb{E}_{\lambda\sim\mathcal{D}_\lambda}[a_\lambda]\sum_{i=1}^{n} q(\mathbf{x}_i)$$

$$q(\mathbf{x}_i) = (g(f_\theta(\mathbf{x}_i)) - \mathbf{y}_i)^\top \mathbf{J}_{f_\theta}(\mathbf{x}_i)(\mathbb{E}[\mathbf{x}'] - \mathbf{x}_i).$$

# Proposed MixupE

The first-order directional derivative is captured by

$$D^1_{\theta,S} := \frac{1}{n}\mathbb{E}_{\lambda\sim\mathcal{D}_\lambda}[a_\lambda]\sum_{i=1}^{n} q(\mathbf{x}_i)$$

$$q(\mathbf{x}_i) = (g(f_\theta(\mathbf{x}_i)) - \mathbf{y}_i)^\top \mathbf{J}_{f_\theta}(\mathbf{x}_i)(\mathbb{E}[\mathbf{x}'] - \mathbf{x}_i).$$

Unfortunately, computing Jacobian in deep models at each step is expensive. We can approximate $q(\mathbf{x}_i)$,

$$q(\mathbf{x}_i) \approx \hat{q}(\mathbf{x}_i) = (\mathbf{y}_i - g(f_\theta(\mathbf{x}_i)))^\top f_\theta(\mathbf{x}_i), \tag{1}$$

- Normalization : $\mathbb{E}_{\mathbf{x}'\sim\mathcal{D}_X}[\mathbf{x}'] = 0$
- ReLU : $\mathbf{J}_{f_\theta}(\mathbf{x}_i)\mathbf{x}_i \approx f_\theta(\mathbf{x}_i)$

# Proposed MixupE

To avoid negativity, the regularization will be

$$R(\theta, S) = \frac{\mathbb{E}_{\lambda \sim \mathcal{D}_\lambda}[a_\lambda]}{n} \sum_{i=1}^{n} |\hat{q}(\mathbf{x}_i)|.$$

# Proposed MixupE

To avoid negativity, the regularization will be

$$R(\theta, S) = \frac{\mathbb{E}_{\lambda \sim \mathcal{D}_\lambda}[a_\lambda]}{n} \sum_{i=1}^{n} |\hat{q}(\mathbf{x}_i)|.$$

Then, the final (normalized) loss will be

$$\mathcal{L}(\theta, S) := \hat{\eta} \left( L_n^{mix}(\theta, S) + \eta R(\theta, S) \right),$$
$$\hat{\eta} = \frac{|L_n^{mix}(\theta, S)|}{|L_n^{mix}(\theta, S) + \eta R(\theta, S)|},$$

where $\hat{\eta}$ is a scaling factor that depends on the magnitudes of $L_n^{mix}(\theta, S)$ and $R(\theta, S)$.

# MixupE Implementation

For each iteration,

1. Sample $\lambda \sim \text{Beta}(\alpha, \beta)$
2. Mixup data with $\tilde{X}, \tilde{Y} \leftarrow \lambda(X, Y) + (1-\lambda)\text{Permute}(X, Y)$
3. Mixup Loss $L_n^{mix}(\theta, X) = \ell(f_\theta(\tilde{X}), \tilde{Y})$
4. Compute first-order directional derivatives that $\hat{q}(X) = f_\theta(X) \otimes (Y - \text{Softmax}(f_\theta(X)))$
5. Get additional loss $R(\theta, X) = \frac{\mathbb{E}_{\lambda \sim \mathcal{D}_\lambda}[a_\lambda]}{n} \sum_{i=1}^{n} |\hat{q}(\mathbf{x}_i)|$
6. Total loss $\mathcal{L}(\theta, S) = \hat{\eta}\left(L_n^{mix}(\theta, S) + \eta R(\theta, S)\right)$

# Generalization Guarantee

- GLM [Zhang et al., 2020]: $h(f_\theta(\mathbf{x})) = A(\theta^\top \mathbf{x})$
- Constraint $\Theta = \{\mathbf{x} \to f_\theta(\mathbf{x}) | \sup_{\mathbf{x}} |\hat{q}(\mathbf{x})| \leq \gamma\}$.
- Expected risk of MixupE: $\tilde{\mathcal{L}}(\theta) = \mathbb{E}_S \mathcal{L}(\theta, S)$

# Generalization Guarantee

- GLM [Zhang et al., 2020]: $h(f_\theta(\mathbf{x})) = A(\theta^\top \mathbf{x})$
- Constraint $\Theta = \{\mathbf{x} \to f_\theta(\mathbf{x}) | \sup_{\mathbf{x}} |\hat{q}(\mathbf{x})| \leq \gamma\}$.
- Expected risk of MixupE: $\tilde{\mathcal{L}}(\theta) = \mathbb{E}_S \mathcal{L}(\theta, S)$

**Theorem 2**

Suppose $A(\cdot)$ is $L_A$-Lipchitz, $\mathcal{X}, \mathcal{Y}$ and $\Theta$ are all bounded, then exist constants $B > 0$, such that for all $\theta \in \Theta$, we have

$$\tilde{\mathcal{L}}(\theta) \leq \hat{\eta} L_n^{mix}(\theta, S) + \frac{2\hat{\eta}\eta L_A \gamma \mathcal{X}}{\sqrt{n}(1 + L_A)} + B\sqrt{\frac{\log(1/\delta)}{2n}} \tag{2}$$

with probability at least $1 - \delta$.

## Generalization Guarantee

- GLM [Zhang et al., 2020]: $h(f_\theta(\mathbf{x})) = A(\theta^\top \mathbf{x})$
- Constraint $\Theta = \{\mathbf{x} \to f_\theta(\mathbf{x}) | \sup_{\mathbf{x}} |\hat{q}(\mathbf{x})| \leq \gamma\}$.
- Expected risk of MixupE: $\tilde{\mathcal{L}}(\theta) = \mathbb{E}_S \mathcal{L}(\theta, S)$

**Theorem 2**

Suppose $A(\cdot)$ is $L_A$-Lipchitz, $\mathcal{X}, \mathcal{Y}$ and $\Theta$ are all bounded, then exist constants $B > 0$, such that for all $\theta \in \Theta$, we have

$$\tilde{\mathcal{L}}(\theta) \leq \hat{\eta} L_n^{mix}(\theta, S) + \frac{2\hat{\eta}\eta L_A \gamma \mathcal{X}}{\sqrt{n}(1 + L_A)} + B\sqrt{\frac{\log(1/\delta)}{2n}} \tag{2}$$

with probability at least $1 - \delta$.

$$\hat{\Theta} : \{\|\theta\|_2^2 \leq \xi\} \Rightarrow \mathcal{R}(\hat{\Theta}, S) = \mathbb{E}_\epsilon \sup_{\|\mathbf{x}_i\|_2^2 \leq \mathcal{X}} \frac{1}{n} \sum_{i=1}^{n} \epsilon_i \theta^\top \mathbf{x}_i \leq \frac{\sqrt{\xi \mathcal{X}}}{\sqrt{n}}$$

# Outline

# Image Classification Test Error (%)

| PreActResNet50 | CIFAR10 | CIFAR100 | SVHN |
|---|---|---|---|
| ERM | 4.71$_{\pm 0.062}$ | 24.68$_{\pm 0.349}$ | 2.80$_{\pm 0.201}$ |
| Mixup | 4.53$_{\pm 0.041}$ | 23.03$_{\pm 0.471}$ | 2.65$_{\pm 0.017}$ |
| *MixupE* | **3.53**$_{\pm 0.047}$ | **20.23**$_{\pm 0.507}$ | **2.42**$_{\pm 0.021}$ |
| PreActResNet101 | | | |
| ERM | 4.21$_{\pm 0.069}$ | 23.20$_{\pm 0.362}$ | 2.95$_{\pm 0.019}$ |
| Mixup | 4.43$_{\pm 0.049}$ | 23.05$_{\pm 0.383}$ | 2.79$_{\pm 0.015}$ |
| *MixupE* | **3.35**$_{\pm 0.049}$ | **18.86**$_{\pm 0.376}$ | **2.35**$_{\pm 0.019}$ |
| Wide-Resnet-28-10 | | | |
| ERM | 4.24$_{\pm 0.101}$ | 22.20$_{\pm 0.108}$ | 2.82$_{\pm 0.049}$ |
| Mixup | 3.03$_{\pm 0.091}$ | 19.38$_{\pm 0.113}$ | 2.48$_{\pm 0.117}$ |
| *MixupE* | **2.94**$_{\pm 0.048}$ | **17.12**$_{\pm 0.111}$ | **2.29**$_{\pm 0.168}$ |

# Other tasks

Table 4: Classification Test Error (%) on tabular datasets from UCI repository. Results are averaged over five trials.

| Dataset | Method | | |
|---|---|---|---|
| | ERM | Mixup | *MixupE* |
| Arrhythmia | **34.60**$_{\pm 3.10}$ | 35.49$_{\pm 3.88}$ | 34.85$_{\pm 3.99}$ |
| Letter | 4.56$_{\pm 0.27}$ | **3.71**$_{\pm 0.18}$ | 4.04$_{\pm 0.20}$ |
| Balance-scale | 3.87$_{\pm 1.03}$ | 3.70$_{\pm 1.00}$ | **3.68**$_{\pm 0.97}$ |
| Mfeat-factors | 2.74$_{\pm 0.81}$ | **2.44**$_{\pm 0.42}$ | 2.56$_{\pm 0.64}$ |
| Mfeat-fourier | 17.69$_{\pm 1.76}$ | 17.80$_{\pm 1.56}$ | **17.57**$_{\pm 1.60}$ |
| Mfeat-karhunen | 3.74$_{\pm 0.58}$ | 3.06$_{\pm 0.29}$ | **2.47**$_{\pm 0.32}$ |
| Mfeat-morph | 25.00$_{\pm 2.10}$ | **24.62**$_{\pm 1.83}$ | 24.66$_{\pm 1.30}$ |
| Mfeat-zernike | 17.58$_{\pm 1.72}$ | **15.19**$_{\pm 1.73}$ | 15.55$_{\pm 0.62}$ |
| CMC | 45.77$_{\pm 1.49}$ | 46.67$_{\pm 1.83}$ | **45.42**$_{\pm 2.05}$ |
| Optdigits | 1.48$_{\pm 0.19}$ | **1.15**$_{\pm 0.21}$ | 1.33$_{\pm 0.14}$ |
| Pendigits | 1.03$_{\pm 0.25}$ | 0.76$_{\pm 0.19}$ | **0.72**$_{\pm 0.16}$ |
| Iris | 9.06$_{\pm 7.01}$ | 8.14$_{\pm 6.48}$ | **7.29**$_{\pm 6.95}$ |
| Mnist_784 | 2.83$_{\pm 0.11}$ | 2.57$_{\pm 0.05}$ | **2.56**$_{\pm 0.14}$ |
| Abalone | 35.05$_{\pm 0.61}$ | 35.07$_{\pm 0.69}$ | **34.91**$_{\pm 0.70}$ |
| Volkert | 33.26$_{\pm 0.62}$ | 32.74$_{\pm 0.76}$ | **32.54**$_{\pm 0.61}$ |

Table 5: Classification Test Error (%) on Google Speech Command Dataset [Warden, 2018]. We run each experiment five times

| Architecture | Method | | |
|---|---|---|---|
| | ERM | Mixup | *MixupE* |
| LeNet | 10.43$_{\pm 0.052}$ | 10.12$_{\pm 0.041}$ | **10.02**$_{\pm 0.042}$ |
| VGG-11 | 6.04$_{\pm 0.059}$ | 4.63$_{\pm 0.047}$ | **3.93**$_{\pm 0.050}$ |
| VGG-13 | 5.77$_{\pm 0.053}$ | 4.68$_{\pm 0.039}$ | **3.84**$_{\pm 0.040}$ |

Table 6: Classification Test Error (%) on graph datasets from the TUDatasets benchmark when following the setup of Xu et al. [2018]. Results are obtained from 10-fold validation.

| Dataset | Method | | |
|---|---|---|---|
| | ERM | Mixup | *MixupE* |
| MUTAG | 10.15$_{\pm 0.06}$ | 10.67$_{\pm 0.05}$ | **10.06**$_{\pm 0.06}$ |
| NCI1 | 17.79$_{\pm 0.02}$ | 18.59$_{\pm 0.02}$ | **17.74**$_{\pm 0.01}$ |
| PTC | 38.37$_{\pm 0.09}$ | **34.87**$_{\pm 0.08}$ | 35.50$_{\pm 0.08}$ |
| PROTEINS | 25.43$_{\pm 0.04}$ | 24.44$_{\pm 0.04}$ | **23.72**$_{\pm 0.04}$ |
| IMDBBINARY | 25.60$_{\pm 0.03}$ | 25.30$_{\pm 0.03}$ | **25.20**$_{\pm 0.03}$ |
| IMDBMULTI | 50.33$_{\pm 0.03}$ | 49.27$_{\pm 0.04}$ | **48.53**$_{\pm 0.03}$ |

# Generalization – Stronger regularization

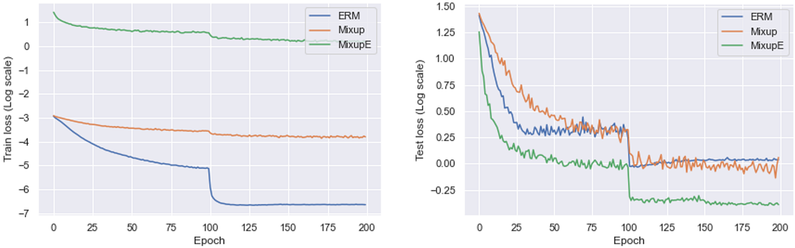



**Figure:** textitMixupE has a higher training loss but lower test loss than Mixup and ERM (Wide-Resnet-28-10).

# Robustness – Generalize to Novel Deformations

**Table:** Test accuracy on novel deformations. All models are trained on normal CIFAR-100.

| **Test Set Deformation** | Mixup | Manifold Mixup | Ours |
|---|---|---|---|
| Rotation $U(-20, 20)$ | 56.48 | 60.08 | **62.23** |
| Rotation $U(-40, 40)$ | 36.78 | 42.13 | **43.08** |
| Shearing $U(-28.6, 28.6)$ | 60.01 | 62.85 | **63.94** |
| Shearing $U(-57.3, 57.3)$ | 39.70 | **44.27** | **43.87** |
| Zoom In (60% rescale) | 13.12 | 11.49 | **15.66** |
| Zoom In (80% rescale) | 50.47 | 52.70 | **54.22** |
| Zoom Out (120% rescale) | 61.62 | **63.59** | 61.39 |
| Zoom Out (140% rescale) | 42.02 | **45.29** | 36.58 |

# Q & A

Thank you!

# references I

Soudry, D., Hoffer, E., Nacson, M. S., Gunasekar, S., and Srebro, N. (2018).
The implicit bias of gradient descent on separable data.
*The Journal of Machine Learning Research*, 19(1):2822–2878.

Zhang, H., Cisse, M., Dauphin, Y. N., and Lopez-Paz, D. (2018).
mixup: Beyond empirical risk minimization.
*International Conference on Learning Representations*.

Zhang, L., Deng, Z., Kawaguchi, K., Ghorbani, A., and Zou, J. (2020).
How does mixup help with robustness and generalization?
In *International Conference on Learning Representations*.